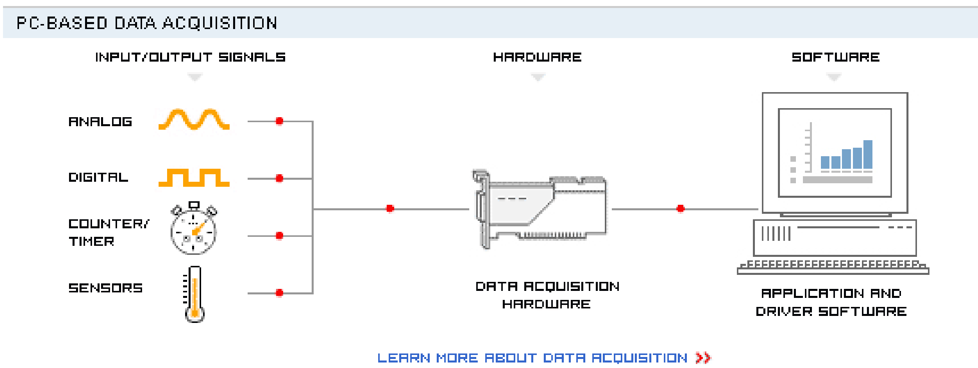
PHY351/651 Laboratory I *Revised 9/1/2009*

Introduction to Labview

Introduction



Modern data acquisition system include three basic elements:

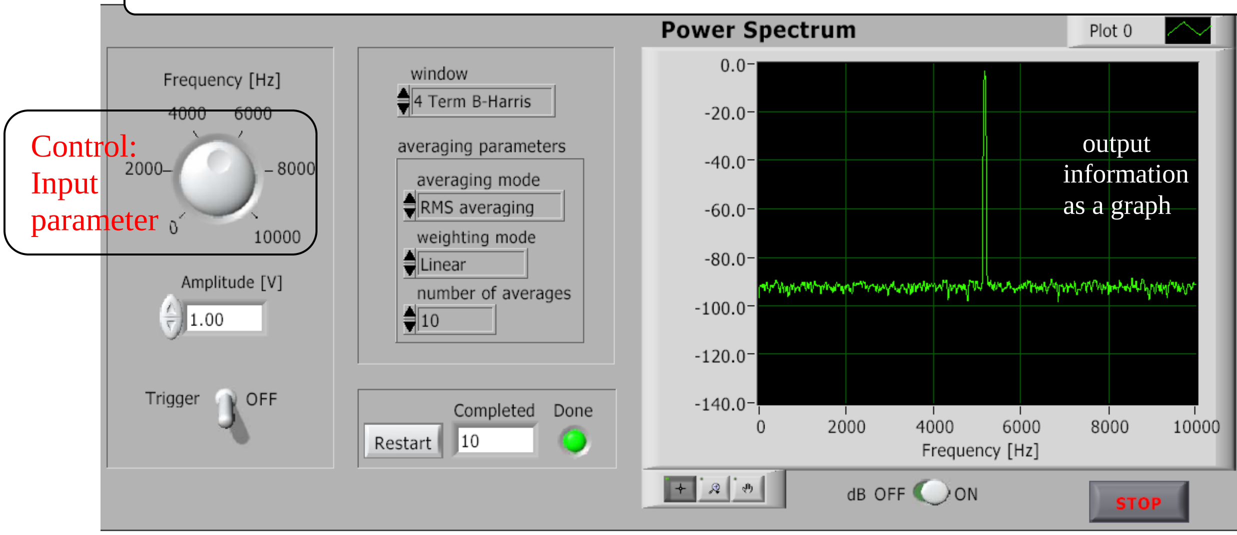
- A sensor transforming a physical property such as temperature, charge, pressure
- A ‘front end electronic’ system, amplifying and shaping the analog signal from the sensor to match the sensitivity and the time properties of the ‘back-end’ data acquisition system.
- A ‘back-end electronics’, further processing the remote signal and transforming it into a ‘digital signal’ [bits that can be manipulated by the on-line computer].
- An ‘on-line computer that takes this information and produces the output information in a manner easily interpreted by the scientist performing the measurement.

Labview provides a programming environment with standard interfaces to a variety of ‘back-end’ electronics. The programming style is very intuitive, it is a “graphical programming”: the program is created using graphics. In addition to be useful for data acquisition, Labview can be used for simulation and general programming as well.

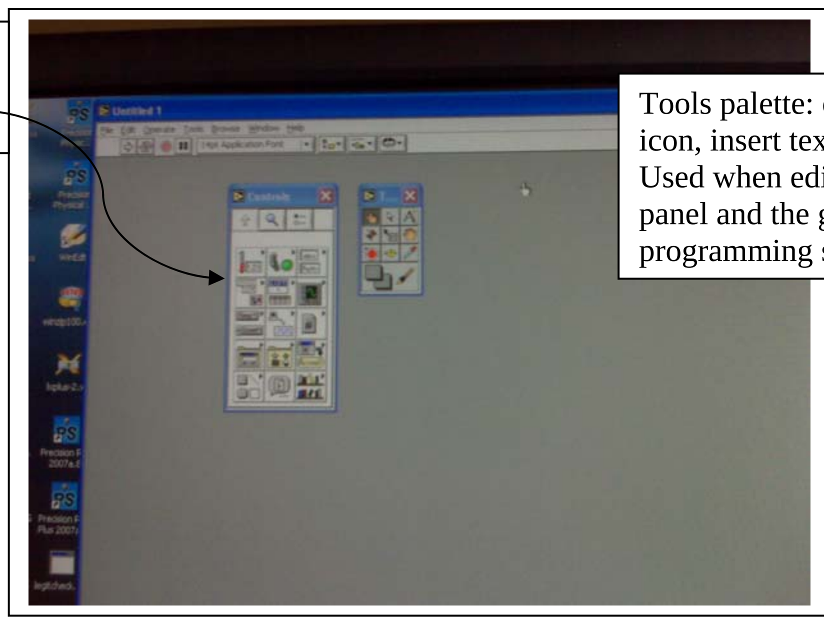
The name LABVIEW is an acronym for Laboratory Virtual Instrument Engineering Workbench. With this tool, you can design measurement and automation systems that employ low-cost, flexible PC technology. LabVIEW's graphical programming language provides easy means for scientists and engineers of diverse background to quickly design and implement complex test and measurement and automation applications. The first cycle of laboratory session will familiarize you with this programming tool and the way it is used in most laboratories.

Introduction to LabVIEW terms:

The VI Front panel (GUI, will be used to run the program and display results)

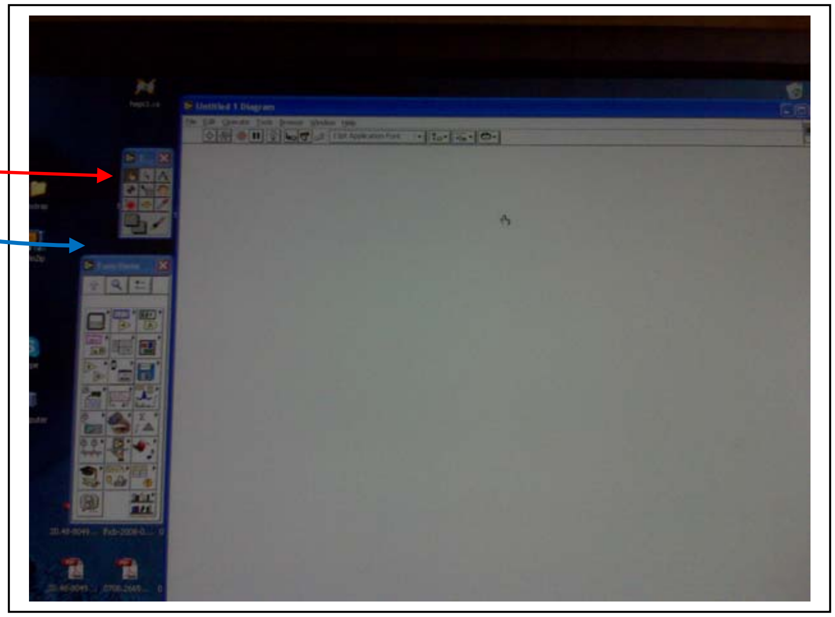


Palette to insert control (input constants) or indicators (output data)



Tools palette: operate, move icon, insert text, wiring tool… Used when editing the front panel and the graphical programming screen

The diagram window: where you write your program
You use the tool menu (wiring tool to connect functions) and the functions menu to add elements. Note: the function arrangement is slightly different in Labview 6i than in the version of labview used in the textbook



ACTIVITY I – INTRODUCTION TO VIRTUAL INSTRUMENTS

Goal: Familiarize yourself with the PC based data acquisition system that you will work with this semester.

Your computer is equipped with NI PCI 6023E or 6024E DAQ board that includes interfaces for multi-channel analog I/O and multi-channel digital I/O functions (you can find out more about these cards in the web page for the class under references). In particular, you can digitize up to 8 differential analog inputs or 16 single-ended analog inputs and generate algorithms that use 8 lines of digital I/O. The 6024E includes 2 lines of analog output as well, and this feature will be used in some of the special projects. We will explore the properties of these boards in more detail in a subsequent lab session. The boards are connected with BNC-2120 connector accessories: they include BNC connectors for analog signals, a variety of interconnections for digital signals and other features that we will explore in future labs. For the time being we will take a look at how an analog signal can be connected to a BNC-2120 input and visualized with a Labview virtual instrument [a labview program].

In order to explore the elements involved in this process, please follow the steps below:

1. Open the Labview icon in your desktop
2. Click on DAQ solution
3. You will be offered the option of configuring your DAQ system in a pop-up window: click on the DAQ configuration utility (MAX).
4. Open the "data neighborhood folder": if you see icons, it means that somebody already defined analog or digital channels: righ-click on the channels and delete them.
5. Configure 2 analog channels to be single-ended referenced voltage channels [ch0 and ch1].
6. Go to solution galleries and select in Bench-top instruments 2 channel oscilloscope.
7. Connect the sine/triangle waveform from the BNC-2120 to one input channel and the TTL waveform in the BNC-2120 interface to another input channel.
8. The 2 Channel Oscilloscope VI should appear. Run it [clicking on the arrow on the left-top corner of the GUI]. You should see the two waveforms. **Stop the execution and write down the period and amplitude of the waveforms measured. Relate them to the parameters that you read in your waveform generator.**
9. Familiarize yourself with the various labview windows.
    a. The GUI that you see is the **FRONT PANEL**, so called because it simulates the front panel of a real instrument [in this case an oscilloscope]. This screen is the one that will be used to make the desired measurement, once the algorithm is implemented.
    b. There are a couple of pop-up windows associated with it:
        i. one is called **CONTROLS**, featuring various elements that you may want to put on the front panel,

ii. the other is called **TOOLS**, and it includes control features that are used both when you modify the front panel and when you modify the block diagram.

c. Now go to the Window control, on the top right, and click on SHOW DIAGRAM: another window will appear, as well as another pop-up menu: the **FUNCTION** menu. The diagram view is the programming view. Here you can see what is meant by graphical programming, as well as the various tools that are used. The programs link inputs and outputs defined in the front panel with various actions that flow from left to right utilizing as basic building blocks the Functions, available in the **Functions Panel**, or SUBVI that are either available in standard libraries or that you will program. Please take some time to explore the **Functions Panel**, in particular locate the Numeric, String, Comparison, Time & Dialog, and Data Acquisition subsections.

10. When you are done, select CLOSE from the file menu both for the diagram window and for the control panel window. BE SURE TO ALWAYS SAY "NO TO ALL", WHEN YOU CLOSE THE CONTROL PANEL OF A VI THAT YOU DID NOT WRITE.

Optional Activity: From the Labview start-up panel, select the option SEARCH EXAMPLES, and explore additional sample VIs: you can learn a lot by observing how these programs are constructed.

For your lab report:
Describe the elements of a data acquisition system, possibly making reference to a system that you are familiar with (from direct experience through some research activity or surfing the web).
Sketch the components of labview and your own data acquisition system drawn from the experience with the analog scope VI.

ACTIVITY II: BUILD YOUR FIRST VI

Goal: Generate a VI that calculates the sum of the squares of two numbers and displays the result.

Steps:
1. Create a folder on the desktop with the name of your team.
1. Open the Front panel and add 2 digital controls [to enter input numbers] and one digital indicator [output number]
2. Use save as in the file menu to save this vi with your favorite name in the folder that you just created.
3. Open the Diagram window and add the necessary numeric functions [hint: define y^2 as yXy] and connect them appropriately with the wiring tool. [the little spool in the tools panel. In the beginning wiring may be a little tricky, so be patient and be sure, at the end of this step, to hit CRTL-B to remove 'broken-wires'.

4. Go back to the Front panel, select Operate from the tool panel and try to run this VI.
5. Now try to make this VI neat: you can change the color with the paintbrush tool, add decorations with the “decorations” tool in the control panel, and add some comments with the text tool in the Tool pop-up window.
6. Run the VI to check that it performs the desired function.
7. On the file menu select VI properties. In the category Documentation, enter your description of this VI. Check the memory usage with the corresponding category. View the location where the VI is stored and other important bookkeeping information in the GENERAL category.
8. Save the complete VI in your group folder.

For your lab report:

Include a picture of the front panel and of the diagram window of your VI with annotations whenever necessary.

ACTIVITY III: BUILD YOUR SECOND VI

Goal: Build a VI that compares an integer number to the number 5 and produces a green LED if the number is greater than 5, as well as a word than is UNDER or OVER depending upon the relationship between the number that you enter and the number 5. Again make this VI as fancy as you can and document it, and then save it in your group folder.

Note: this time I am not giving you step by step instructions, but, following the steps above and thinking a bit about the best modules to use in this exercise, you should be able to succeed.

For your lab report:

Include a picture of the front panel and of the diagram window of your VI with annotations whenever necessary.

ACTIVITY IV: Creating your first SUB-VI

Goal: Transforming the previous VI into a sub-VI that can be used in more complex program.

If you have any experience with programming, you are well aware that good programs are modular: you design your building blocks performing a specific function and then you link them together as needed in a more complex program. If you go back to the example VI that you explored in activity I, you will see that there are several icons that you can open that represent sub-VIs.

In order to transform a VI into a subVI that can be used in other programs you have to learn to manipulate the ICON [on the top right corner of your VI]. You can edit it starting from the VI property window or double-clicking it. Change the icon to be a simple string with a color background.

Now you need to be able to connect this VI with the outside world. Follow these steps to assign a terminal to a control or an indicator.

1. On the front panel view, pop up the icon panel and select show connector.
2. Click on a terminal: the wiring tool will appear. Connect it to the desired indicator/control element. A moving dotted line will appear when you successfully make the connection.
3. Click on an open area of the front panel.
4. Repeat step 2 as necessary.

Save this VI with another name and **construct a VI that performs the same function utilizing it as a sub-vi.** Document this VI and save it in your group folder.

For your lab report:

Include a picture of the front panel and of the diagram window of your VI with annotations whenever necessary.

ACTIVITY V

Follow the example in Ch. I of your textbook and buid a VI that displays a sine waveform.

ACTIVITY VI

Write a program called **Iterations Until Integer Equal Five**, which randomly generates an integer in the range from 1 to 10, and iterates this random process until the integer equals 5. The number of iterations required until the randomly generated integer equals 5 is then displayed in the front panel in a numeric indicator which is labeled **Required Iterations.**

For your lab report:

Include a picture of the front panel and of the diagram window of your VI with annotations whenever necessary.